

CLAIMS

What is claimed is:

Sub A1  
1. A function object for use in creating a mapping in a mapping tool with a graphical user interface, between a source object having a source object node and a target object having a target object node, comprising:

a script component having computer-executable instructions for performing a function;

a graphical component associated with the function, having an input and an output, and adapted to allow a user to graphically associate the input with a source object node and to associate the output with a target object node in the graphical user interface; and

an interface component having a globally unique identifier and adapted to provide the script component to a compiler in the mapping tool and to provide the graphical component to the graphical user interface.

2. The function object of claim 1, wherein the interface component further includes a category identifier component.

3. The function object of claim 1, wherein the interface component further comprises a class identifier component.

4. The function object of claim 1, wherein the interface component further comprises a function category identifier component associated with the function.

5. The function object of claim 1, wherein the function category identifier component includes one of string, mathematical, logical, date, conversion, scientific, advanced, and custom.

6. The function object of claim 1, further comprising a second interface component adapted to allow a user to drag and drop the graphical component in the graphical user interface.

7. The function object of claim 5, wherein the second interface component is associated with the function object by the mapping tool.

8. The function object of claim 1, further comprising:  
a plurality of script components having computer-executable instructions for performing a plurality of functions; and  
a plurality of graphical components individually associated with one of the plurality of functions, individually having an input and an output, and individually adapted to allow a user to graphically associate the input with a source object node and to associate the output with a target object node in the graphical user interface;  
wherein the interface component is further adapted to provide the plurality of script components to a compiler in the mapping tool and to provide the plurality of graphical components to the graphical user interface.

9. The function object of claim 7, wherein the interface component further comprises a plurality of function category identifier components individually associated with the plurality of functions.

10. The function object of claim 1, wherein at least one of the plurality of function category identifier components includes one of string, mathematical, logical, date, conversion, scientific, advanced, and custom.

11. The function object of claim 10, wherein the plurality of script components having computer-executable instructions for performing a plurality of functions are in a scripting language.

12. The function object of claim 11, wherein the scripting language is one of java script, visual basic script, and visual C++.

13. The function object of claim 1, wherein the script component having computer-executable instructions for performing the function is in a scripting language.

14. The function object of claim 13, wherein the scripting language is one of java script, visual basic script, and visual C++.

15. The function object of claim 1, wherein the function object is a COM object, and the interface component is a COM interface.

16. In a mapping tool with a graphical user interface, a method of creating a mapping between a source object having a source object node and a target object having a target object node, the method comprising:

providing a function object having a script component with computer-executable instructions for performing a function, a graphical component associated with the function and having an input and an output, and an interface component;

displaying the graphical component in the user interface;

graphically associating a source object node with the input using a user interface selection device;

graphically associating a target object node with the output using the user interface selection device; and

creating a mapping including the computer-executable instructions, and operative to perform the function according to the source object node, and to provide an output value associated with the target object node according to the function.

17. The method of claim 16, wherein providing the function object further comprises obtaining the function object from a host computer.

18. The method of claim 17, wherein obtaining the function object from a host computer further comprises obtaining the function object from a DLL file in the host computer.

19. The method of claim 18, wherein obtaining the function object from a DLL file in the host computer further comprises:

searching the DLL files in the host computer;  
identifying the function object in the searched DLL files; and  
loading the function object into the mapping tool.

20. The method of claim 16, wherein providing the function object further comprises obtaining the function object from a global communications network.

21. The method of claim 20, wherein obtaining the function object from a global communications network further comprises obtaining the function object from an Internet web site.

22. The method of claim 21, wherein obtaining the function object from an Internet web site further comprises one of searching the Internet for the function object according to a mapping tool startup script, searching the Internet for the function object according to a user command, and searching a web site provided by the user for the function object.

23. The method of claim 16, further comprising:  
providing a plurality of script components with computer-executable instructions for performing one of a plurality of functions, a plurality of graphical components individually associated with one of the plurality of functions and individually having an input and an output, wherein the interface component is associated with the plurality of script components and the plurality of graphical components; and

displaying at least one of the plurality of graphical components in the user interface.

24. The method of claim 16, wherein providing the function object further comprises creating a wrapper object in the mapping tool, including the interface component of the function object and a second interface component adapted to allow a user to drag and drop the graphical component in the graphical user interface.

25. The method of claim 16, wherein displaying the graphical component further comprises obtaining the graphical component from a function object source via the interface component.

26. The method of claim 25, wherein the function object source is a DLL file on a host computer.

27. The method of claim 25, wherein obtaining the graphical component from a function object source comprises obtaining the graphical component from a DLL file via the Internet.

28. The method of claim 16, wherein displaying the graphical component further comprises:

displaying a function object palette in the graphical user interface;  
displaying the graphical component on the function object palette; and  
allowing a user to drag and drop the graphical component from the function object palette to a mapping screen region in the graphical user interface.

29. The method of claim 28, further comprising:  
providing a plurality of script components with computer-executable instructions for performing one of a plurality of functions, a plurality of graphical components individually associated with one of the plurality of functions and individually having an input and an output, wherein the interface component is

associated with the plurality of script components and the plurality of graphical components;

displaying a function object palette in the graphical user interface;

displaying at least one of the plurality of graphical components on the function object palette; and

allowing a user to drag and drop the at least one of the plurality of graphical components from the function object palette to a mapping screen region in the graphical user interface.

30. The method of claim 16, wherein the mapping tool further comprises a compiler component, and wherein creating the mapping further comprises:

invoking the compiler component to generate compiled output code; and

providing the computer-executable instructions from the script component to the compiler via the interface component.

31. The method of claim 16, wherein the mapping tool further comprises a compiler component, and wherein creating the mapping further comprises:

providing a plurality of script components with computer-executable instructions for performing one of a plurality of functions, a plurality of graphical components individually associated with one of the plurality of functions and individually having an input and an output, wherein the interface component is associated with the plurality of script components and the plurality of graphical components;

displaying a function object palette in the graphical user interface;

displaying at least one of the plurality of graphical components on the function object palette;

allowing a user to drag and drop the at least one of the plurality of graphical components from the function object palette to a mapping screen region in the graphical user interface;

invoking the compiler component to generate compiled output code; and

providing the computer-executable instructions from the plurality of script components to the compiler via the interface component.

32. A mapping tool with a graphical user interface for creating a mapping between a source object having a source object node and a target object having a target object node, comprising:

means for providing a function object having a script component with computer-executable instructions for performing a function, a graphical component associated with the function and having an input and an output, and an interface component;

a display device in the graphical user interface adapted to display the graphical component in the user interface;

a user interface selection device adapted to graphically associate a source object node with the input, and to graphically associate a target object node with the output; and

a compiler adapted to create a mapping including the computer-executable instructions, and operative to perform the function according to the source object node, and to provide an output value associated with the target object node according to the function.

33. In a mapping tool with a graphical user interface, a method of creating a function object for use in creating a mapping between a source object and a target object, the method comprising:

creating a script component having computer-executable instructions for performing a function using the user interface;

creating a graphical component associated with the function and having an input and an output;

creating an interface component adapted to provide the script component to a compiler in the mapping tool and to provide the graphical component to the graphical user interface; and

associating the script component, the graphical component, and the interface component.

34. The method of claim 33, wherein creating the script component further comprises receiving a user-defined text file including the computer-executable instructions, and creating the script component using the computer-executable instructions from the text file.

35. The method of claim 34, wherein the computer-executable instructions include one of basic, visual basic, VB script, C++, visual C++, java, java script, and Perl .

36. The method of claim 33, wherein creating the script component comprises receiving function information from a user, and creating the computer-executable instructions based on the function information.

37. The method of claim 33, wherein creating the interface component comprises:

receiving a text file from a user, wherein the text file includes information related to the function; and  
creating the interface component according to the information in the text file.

38. The method of claim 37, wherein the information related to the function is in XML.

39. The method of claim 33, further comprising:  
prompting a user for information related to the function object;  
receiving prompted information from the user via the graphical user interface;  
creating the script component having computer-executable instructions for performing a function using the prompted information; and



creating the graphical component associated with the function and having an input and an output using the prompted information.

40. The method of claim 39, wherein prompting a user for information related to the function object comprises providing a wizard in the graphical user interface.

41. A function object creation tool for creating a function object for use in creating a mapping in a mapping tool between a source object and a target object, the function object creation tool comprising:

means for creating a script component having computer-executable instructions for performing a function using the user interface;

means for creating a graphical component associated with the function and having an input and an output;

means for creating an interface component adapted to provide the script component to a compiler in the mapping tool and to provide the graphical component to the graphical user interface; and

means for associating the script component, the graphical component, and the interface component.

42. A computer-readable medium having computer-executable instructions for performing the following steps:

providing a function object having a script component with computer-executable instructions for performing a function, a graphical component associated with the function and having an input and an output, and an interface component;

displaying the graphical component in a user interface;

graphically associating a source object node with the input using a user interface selection device;

graphically associating a target object node with the output using the user interface selection device; and

creating a mapping including the computer-executable instructions, and operative to perform the function according to the source object node, and to provide an output value associated with the target object node according to the function.

43. A computer-readable medium having computer-executable instructions for performing the following steps:

creating a script component having computer-executable instructions for performing a function;

creating a graphical component associated with the function and having an input and an output;

creating an interface component adapted to provide the script component to a compiler in a mapping tool and to provide the graphical component to a graphical user interface; and

associating the script component, the graphical component, and the interface component.

44. A function object for use in creating a mapping, between a source object having a source object node and a target object having a target object node, comprising:

a script component having computer-executable instructions for performing a function;

a graphical component associated with the function, having an input and an output, and adapted to allow a user to graphically associate the input with a source object node and to associate the output with a target object node; and

an interface component having a globally unique identifier and adapted to provide the script component to a compiler and to provide the graphical component to a graphical user interface.

45. A computer-readable medium having computer-executable instructions for performing the following steps:

creating a script component;  
 creating a function;  
 creating a graphical component;  
 creating an output;  
 creating an interface component;  
 and to provide the graphical user interface;  
 associating the script component with the graphical user interface.

creating a graphical component associated with the function and having an

creating an interface component adapted to provide the script component to a

associating the script component, the graphical component, and the interface